App ndix A

A Source Code Compendium of System Accessible COIGN Definitions

```
/////////////////////////////////////////////////////////////////////////////////
//
//    Module:            coign
//    File:        coign.h
//    Author:            Galen C. Hunt
//    Copyright:   1996-1998
//    Abstract:    System accessible Coign definitions.
//
#pragma once
#ifndef _COIGN_H_
#define _COIGN_H_

#define NODEBUG

/////////////////////////////////////////////////////////////////////////////////
//
#include <objbase.h>
#include <objidl.h>
#include <ocidl.h>
#include <olectl.h>
#include "coignidl.h"
#ifdef INITGUID
#include "coignidl_i.c"
#endif

#include "coignusr.h"

///////////////////////////////////////////////////////////// .coign Section Header.
//
#pragma pack(push, 16)

struct CCoignPayloadHeader
{
      ULONG        dwFlags;
      CLSID        LoggerClsid;
      CLSID        InformerClsid;
      CLSID        ClassifierClsid;
      CLSID        OtherClsid;
};

#pragma pack(pop)

/////////////////////////////////////////////////////////////////////////////////
//
DEFINE_GUID(CLSID_CoignNull,

      0x00000000,0x0000,0x0000,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00);

DEFINE_GUID(GUID_CoignPayloadHeader,                                     // Payload
Header

      0x9ceeb00f,0xe415,0x11d0,0x98,0xd1,0x00,0x60,0x97,0xb0,0x10,0xe3);
DEFINE_GUID(CLSID_CoignLoggerNull,                                       // Null Logger
      0x9ceeb014,0xe415,0x11d0,0x98,0xd1,0x00,0x60,0x97,0xb0,0x10,0xe3);
DEFINE_GUID(CLSID_CoignLoggerEvent,                             // Event
Logger

      0x9ceeb015,0xe415,0x11d0,0x98,0xd1,0x00,0x60,0x97,0xb0,0x10,0xe3);
DEFINE_GUID(CLSID_CoignLoggerProfile,                           // Profile
Logger

      0x9ceeb016,0xe415,0x11d0,0x98,0xd1,0x00,0x60,0x97,0xb0,0x10,0xe3);

DEFINE_GUID(CLSID_CoignInformerNull,                            // Null
Informer

      0x9ceeb018,0xe415,0x11d0,0x98,0xd1,0x00,0x60,0x97,0xb0,0x10,0xe3);
DEFINE_GUID(CLSID_CoignInformerNdr,                             // NDR
Informer

      0x9ceeb019,0xe415,0x11d0,0x98,0xd1,0x00,0x60,0x97,0xb0,0x10,0xe3);
DEFINE_GUID(CLSID_CoignInformerRegistry,                  // Register-based
Informer

      0x9ceeb01a,0xe415,0x11d0,0x98,0xd1,0x00,0x60,0x97,0xb0,0x10,0xe3);

DEFINE_GUID(CLSID_CoignClassifierNull,                          // Null
Classifier

      0x9ceeb01c,0xe415,0x11d0,0x98,0xd1,0x00,0x60,0x97,0xb0,0x10,0xe3);
DEFINE_GUID(CLSID_CoignClassifierClsid,                   // CLSID Classifier

      0x9ceeb01d,0xe415,0x11d0,0x98,0xd1,0x00,0x60,0x97,0xb0,0x10,0xe3);
DEFINE_GUID(CLSID_CoignClassifierClsidFunc,               // CLSID-Function
Classifier

      0x9ceeb01e,0xe415,0x11d0,0x98,0xd1,0x00,0x60,0x97,0xb0,0x10,0xe3);
DEFINE_GUID(CLSID_CoignClassifierStack,                   // Stack Classifier

      0x9ceeb01f,0xe415,0x11d0,0x98,0xd1,0x00,0x60,0x97,0xb0,0x10,0xe3);
DEFINE_GUID(CLSID_CoignClassifierObjectStack,             // Object-Stack
Classifier

      0x9ceeb020,0xe415,0x11d0,0x98,0xd1,0x00,0x60,0x97,0xb0,0x10,0xe3);
DEFINE_GUID(CLSID_CoignClassifierLog,                           // Logging
Classifier

      0x9ceeb021,0xe415,0x11d0,0x98,0xd1,0x00,0x60,0x97,0xb0,0x10,0xe3);
DEFINE_GUID(CLSID_CoignClassifierEP3C,                          // EP3C
Classifier

      0x9ceeb022,0xe415,0x11d0,0x98,0xd1,0x00,0x60,0x97,0xb0,0x10,0xe3);
DEFINE_GUID(CLSID_CoignClassifierI3C,                           // I3C
Classifier

      0x9ceeb023,0xe415,0x11d0,0x98,0xd1,0x00,0x60,0x97,0xb0,0x10,0xe3);
DEFINE_GUID(CLSID_CoignClassifierIncremental,             // Incremental
Classifier
```

```
        0x9ceeb024,0xe415,0x11d0,0x98,0xd1,0x00,0x60,0x97,0xb0,0x10,0xe3);

DEFINE_GUID(CLSID_CoignClassificationDepth,                          //
Classification Depth

        0x9ceeb02f,0xe415,0x11d0,0x98,0xd1,0x00,0x60,0x97,0xb0,0x10,0xe3);

DEFINE_GUID(CLSID_CoignObjectPartition,                          // Partition
Assignments

        0x9ceeb030,0xe415,0x11d0,0x98,0xd1,0x00,0x60,0x97,0xb0,0x10,0xe3);

DEFINE_GUID(CLSID_CoignRemoteHost,

        0x9ceeb050,0xe415,0x11d0,0x98,0xd1,0x00,0x60,0x97,0xb0,0x10,0xe3);
DEFINE_GUID(CLSID_CoignLocalHost,

        0x9ceeb051,0xe415,0x11d0,0x98,0xd1,0x00,0x60,0x97,0xb0,0x10,0xe3);

DEFINE_GUID(CLSID_CoignSystem,

        0x9ceeb040,0xe415,0x11d0,0x98,0xd1,0x00,0x60,0x97,0xb0,0x10,0xe3);
DEFINE_GUID(CLSID_CoignApplication,

        0x9ceeb041,0xe415,0x11d0,0x98,0xd1,0x00,0x60,0x97,0xb0,0x10,0xe3);
DEFINE_GUID(CLSID_CoignChannelHookExtension,

        0x9ceeb042,0xe415,0x11d0,0x98,0xd1,0x00,0x60,0x97,0xb0,0x10,0xe3);

DEFINE_OLEGUID(IID_IStdIdentity, 0x0000001bL, 0, 0);

DEFINE_GUID(IID_IRpcChannelBuffer2,

        0x594f31d0,0x7f19,0x11d0,0xb1,0x94,0x00,0xa0,0xc9,0x0d,0xc8,0xbf);

DEFINE_GUID(CLSID_StgDocFile,

        0xbeef0001,0xe912,0x11d1,0xb6,0xcc,0x00,0x60,0x97,0xb0,0x10,0xe3);
DEFINE_GUID(CLSID_DataAdviseHolder,

        0xbeef0002,0xe912,0x11d1,0xb6,0xcc,0x00,0x60,0x97,0xb0,0x10,0xe3);

///////////////////////////////////////////////////////////////////////////////
//
enum {
        COIGN_FLAG_IS_REMOTE_SURROGATE          = 0x80000000,    //
coignrem.exe

        COIGN_FLAG_RI_FLAG_IN                   = 0x8000,        // Input
Parameter
        COIGN_FLAG_RI_FLAG_OUT                  = 0x4000,        // Output
Parameter
        COIGN_FLAG_RI_FLAG_CONSTANT             = 0x2000,        // Constant
IID Interface
        COIGN_FLAG_RI_FLAG_DEREFERENCE          = 0x1000,        // Ptr to
Interface Ptr
```

```
      COIGN_FLAG_RI_FLAG_OFFSET                  = 0x0fff,          // Offset Mask

      COIGN_FLAG_CLASSIFIER_FLAG_APP             = 0,
      COIGN_FLAG_CLASSIFIER_FLAG_SYS             = 1,
      COIGN_FLAG_CLASSIFIER_FLAG_ROT             = 2,
      COIGN_FLAG_CLASSIFIER_FLAG_UNK             = 3,
      COIGN_FLAG_CLASSIFIER_FLAG_NONE          = 4,
      COIGN_FLAG_CLASSIFIER_FLAG_BASE          = 5,
};

enum {
      COIGN_HOST_LOCAL                     = 0,               // Anynode (here)
      COIGN_HOST_CLIENT                    = 1,               // Home node.
      COIGN_HOST_SERVER                    = 2,               // Other Node.
      COIGN_HOST_LAST                            = 8,               // Max Node.
};

///////////////////////////////////////////////////////////////////////////////
//
#define S_COIGN_CREATED                                ((HRESULT)0x00000010L)
#define     S_COIGN_EXISTED
      ((HRESULT)0x00000011L)
#define S_COIGN_SIZE_REMOTE                            ((HRESULT)0x00000100L)
#define     E_COIGN_IDL_REF_NULL                       ((HRESULT)0x800706f4L)

/////////////////////////////////////////////////////////// Assertion Handling.
//
#ifndef _DEFINED_ASMBREAK_
#define _DEFINED_ASMBREAK_
#define ASMBREAK()      __asm { int 3 }
//#define ASMBREAK()    DebugBreak()
#endif      // _DEFINED_ASMBREAK_

#ifndef NODEBUG
#undef ASSERT
VOID CoignAssertMessage(CONST PCHAR szMsg, CONST PCHAR szFile, ULONG nLine);
#define ASSERT(x)   \
do { if (!(x)) { CoignAssertMessage(#x, __FILE__, __LINE__); ASMBREAK(); }}
while (0)
      ;
#undef ASSERTX
#define ASSERTX(x)   \
do { if (!(x)) { CoignAssertMessage(#x, __FILE__, __LINE__); PCHAR p=(PCHAR)(x);
*p = 1; }} while (0)
      ;
#else // NODEBUG
#undef ASSERT
#define ASSERT(x)
#undef ASSERTX
#define ASSERTX(x)
#endif      // NODEBUG

///////////////////////////////////////////////////////////////////////////////
#ifndef _DEFINED_LOADCYCLECOUNTER_
#define _DEFINED_LOADCYCLECOUNTER_
#define CoignLoadCycleCount(x) \
_asm { \
    __asm lea   ecx, x                  \
    __asm       _emit 0x0f              \
    __asm       _emit 0x31              \
    __asm mov [ecx], eax          \
    __asm mov [ecx+4], edx  \
}
#endif // _DEFINED_LOADCYCLECOUNTER_

#define arrayof(x)              (sizeof(x)/sizeof(x[0]))
#define    wcssize(x)           ((wcslen(x) + 1) * sizeof(WCHAR))
#define    strsize(x)           ((strlen(x) + 1) * sizeof(CHAR))

/////////////////////////////////////////////////////////////////////// Functions.
//
STDAPI CoignEnableRecord(void);
STDAPI CoignDisableRecord(void);
STDAPI CoignTestRecord(void);
STDAPI CoignEnableDebug(void);
STDAPI CoignDisableDebug(void);
STDAPI CoignTestDebug(void);
STDAPI CoignReset(void);
STDAPI CoignFlush(void);
STDAPI CoignDisableTryCatch(VOID);
STDAPI CoignEnableTryCatch(VOID);

STDAPI CoignSetOption(PWCHAR pwzOption, PWCHAR pwzValue);
STDAPI CoignUnwrapInterface(IUnknown *punkSrc, IUnknown **ppunkDst);
STDAPI CoignGetRawInterface(IUnknown *punkSrc, IUnknown **ppunkDst);
STDAPI_(PVOID) CoignToRawInterface(PVOID);
STDAPI_(PVOID) CoignWrapInterface(PVOID);
STDAPI CoignClientComponent(IUnknown *punk);
STDAPI CoignServerComponent(IUnknown *punk);
STDAPI CoignMessage(PCSTR msg, ...);

STDAPI CoignNull(void);
STDAPI CoignTerminate(HINSTANCE hInst);
STDAPI CoignMessageWrite(PBYTE pbData, ULONG cbData);
STDAPI CoignLog(PCSTR msg, ...);
STDAPI CoignLogWrite(PBYTE pbData, ULONG cbData);

STDAPI CoignInternalCodeAdd(PVOID pv);
STDAPI CoignInternalCodeRemove(PVOID pv);
STDAPI CoignInternalCodeFind(PVOID pv);
STDAPI CoignInternalCodeFindRegion(PVOID pv, PVOID *ppBeg, PVOID *ppEnd);
STDAPI CoignSystemCodeAdd(PVOID pv);
STDAPI CoignSystemCodeRemove(PVOID pv);
STDAPI CoignSystemCodeFind(PVOID pv);
STDAPI CoignSystemCodeFindRegion(PVOID pv, PVOID *ppBeg, PVOID *ppEnd);
STDAPI CoignFindExtraData(REFCLSID rclsid, PBYTE *ppbData, ULONG *pcbData);

STDAPI CoignSelectQueryInterface(DWORD dwCount, MULTI_QI *pResults,
                                                  REFIID riid, void **ppv);

STDAPI CoignConnectServer(OLECHAR *pwszHostName /*[in]*/, ULONG *pnHostId);
STDAPI CoignFindFactory(ULONG nHostId, ICoignFactory **ppFactory);
STDAPI CoignNotifyOnClose(HRESULT (*pfNotifyCallback)(VOID));
```

Appendix A

```
STDAPI CoignGetLogger(ICoignLogger **ppLogger);
STDAPI CoignGetInformer(ICoignInterfaceInformer **ppInformer);
STDAPI CoignGetClassifier(ICoignObjectClassifier **ppClassifier);

STDAPI CoignDumpStack(void);
STDAPI CoignGetProcessorCycles(LONGLONG *pllCycles);

//////////////////////////////////////////////////////////////////////////////
//
STDAPI Coign_DllGetClassObject(REFCLSID rclsidReal,
                                             IUnknown * (*pfNew)(VOID),
                                             PLONG pnLock,
                                             REFCLSID rclsidWanted,
                                             REFIID riid,
                                             LPVOID *ppv);
STDAPI Coign_DllRegisterServer(REFCLSID clsid,
                                             HINSTANCE hInst,
                                             PWCHAR pwzDescription);
STDAPI Coign_DllUnregisterServer(REFCLSID clsid);

/////////////////////////////////////////////////////////////////// Rerouting.
//
STDAPI Coign_CoInitialize(LPVOID pvReserved);
STDAPI Coign_CoInitializeEx(LPVOID pvReserved,
                                          DWORD dwCoInit);

STDAPI Coign_CoMarshalInterface(LPSTREAM pStm,
                                                REFIID riid,
                                                LPUNKNOWN pUnk,
                                                DWORD dwDestContext,
                                                LPVOID pvDestContext,
                                                DWORD mshlflags);
STDAPI Coign_CoUnmarshalInterface(LPSTREAM pStm,
                                                  REFIID riid,
                                                  LPVOID FAR* ppv);

STDAPI Coign_CoGetClassObject(REFCLSID rclsid,
                                            DWORD dwClsContext,
                                            PVOID pvReserved,
                                            REFIID riid,
                                            PVOID *ppv);
STDAPI Coign_CoCreateInstance(REFCLSID rclsid,
                                            LPUNKNOWN punkOuter,
                                            DWORD dwClsContext,
                                            REFIID riid,
                                            PVOID *ppv);
STDAPI Coign_CoGetInstanceFromFile(COSERVERINFO *pServerInfo,
                                                   CLSID *pClsid,
                                                   IUnknown *punkOuter,
                                                   DWORD dwClsCtx,
                                                   DWORD grfMode,
                                                   const OLECHAR *pwszName,
                                                   DWORD dwCount,
                                                   MULTI_QI *pResults);
STDAPI Coign_CoGetInstanceFromIStorage(COSERVERINFO *pServerInfo,
                                                         CLSID *pClsid,
                                                         IUnknown *punkOuter,
                                                         DWORD dwClsCtx,
                                                         IStorage *pstg,
                                                         DWORD dwCount,
                                                         MULTI_QI *pResults);
STDAPI Coign_CoCreateInstanceEx(REFCLSID Clsid,
                                                IUnknown *punkOuter,
                                                DWORD dwClsCtx,
                                                COSERVERINFO *pServerInfo,
                                                DWORD dwCount,
                                                MULTI_QI *pResults);
STDAPI Coign_StgCreateDocfile(const OLECHAR *pwcsName,
                                            DWORD grfMode,
                                            DWORD reserved,
                                            IStorage **ppstgOpen);
STDAPI Coign_OleCreate(REFCLSID rclsid,
                                 REFIID riid,
                                 DWORD renderopt,
                                 FORMATETC *pFormatEtc,
                                 IOleClientSite *pClientSite,
                                 IStorage *pStg,
                                 void **ppv);
STDAPI Coign_OleCreateFromData(LPDATAOBJECT pSrcDataObj,
                                             REFIID riid,
                                             DWORD renderopt,
                                             LPFORMATETC pFormatEtc,
                                             LPOLECLIENTSITE pClientSite,
                                             LPSTORAGE pStg,
                                             PVOID *ppv);
STDAPI Coign_OleLoad(IStorage *pStg,
                               REFIID riid,
                               IOleClientSite *pClientSite,
                               PVOID *ppv);
STDAPI Coign_CoRegisterClassObject(REFCLSID rclsid,
                                                   LPUNKNOWN pUnk,
                                                   DWORD dwClsContext,
                                                   DWORD flags,
                                                   LPDWORD lpdwRegister);
STDAPI Coign_CoRevokeClassObject(DWORD dwRegister);
STDAPI Coign_StgCreateDocfileOnILockBytes(ILockBytes *plkbyt,
                                                               DWORD grfMode,
                                                               DWORD reserved,
                                                               IStorage
**ppstgOpen);
STDAPI Coign_StgOpenStorage(const OLECHAR *pwcsName,
                                          IStorage *pstgPriority,
                                          DWORD grfMode,
                                          SNB snbExclude,
                                          DWORD reserved,
                                          IStorage **ppstgOpen);
STDAPI Coign_StgOpenStorageOnILockBytes(ILockBytes *plkbyt,
                                                            IStorage
*pstgPriority,
                                                            DWORD grfMode,
                                                            SNB snbExclude,
                                                            DWORD reserved,
                                                            IStorage
**ppstgOpen);
```

```
STDAPI Coign_StgOpenAsyncDocfileOnIFillLockBytes(IFillLockBytes *pflb,
                                                                          DWORD
grfMode,
                                                                          DWORD
asyncFlags,

IStorage **ppstgOpen);
STDAPI Coign_StgGetIFillLockBytesOnILockBytes(ILockBytes *pilb,

IFillLockBytes **ppflb);
STDAPI Coign_StgGetIFillLockBytesOnFile(const OLECHAR *pwcsName,
                                                             IFillLockBytes
**ppflb);
STDAPI Coign_BindMoniker(LPMONIKER pmk,
                                     DWORD grfOpt,
                                     REFIID riid,
                                     PVOID *ppvResult);
STDAPI Coign_CoGetObject(LPCWSTR pszName,
                                     BIND_OPTS *pBindOptions,
                                     REFIID riid,
                                     void **ppv);
STDAPI Coign_MkParseDisplayName(LPBC pbc,
                                                 LPCOLESTR szUserName,
                                                 ULONG *pchEaten,
                                                 LPMONIKER *ppmk);
STDAPI Coign_MonikerRelativePathTo(LPMONIKER pmkSrc,
                                                    LPMONIKER pmkDest,
                                                    LPMONIKER *ppmk,
                                                    BOOLEAN dwReserved);
STDAPI Coign_MonikerCommonPrefixWith(LPMONIKER pmkThis,
                                                        LPMONIKER pmkOther,
                                                        LPMONIKER *ppmk);
STDAPI Coign_CreateBindCtx(DWORD reserved,
                                       LPBC *ppbc);
STDAPI Coign_CreateGenericComposite(LPMONIKER pmkFirst,
                                                       LPMONIKER pmkRest,
                                                       LPMONIKER *ppmk);
STDAPI Coign_CreateClassMoniker(REFCLSID rclsid,
                                                 LPMONIKER *ppmk);
STDAPI Coign_CreateFileMoniker(LPCOLESTR lpszPathName,
                                              LPMONIKER *ppmk);
STDAPI Coign_CreateItemMoniker(LPCOLESTR lpszDelim,
                                              LPCOLESTR lpszItem,
                                              LPMONIKER *ppmk);
STDAPI Coign_CreateAntiMoniker(LPMONIKER *ppmk);
STDAPI Coign_CreatePointerMoniker(LPUNKNOWN punk,
                                                   LPMONIKER *ppmk);
STDAPI Coign_GetRunningObjectTable(DWORD reserved,
                                                    LPRUNNINGOBJECTTABLE *pprot);
STDAPI Coign_CreateDataAdviseHolder(LPDATAADVISEHOLDER* ppDAHolder);
STDAPI Coign_OleCreateEx(REFCLSID rclsid,
                                     REFIID riid,
                                     DWORD dwFlags,
                                     DWORD renderopt,
                                     ULONG cFormats,
                                     DWORD* rgAdvf,
                                     LPFORMATETC rgFormatEtc,
                                     IAdviseSink* lpAdviseSink,
                                     DWORD* rgdwConnection,
                                     LPOLECLIENTSITE pClientSite,
                                     LPSTORAGE pStg,
                                     PVOID* ppv);
STDAPI Coign_OleCreateFromDataEx(LPDATAOBJECT pSrcDataObj,
                                                 REFIID riid,
                                                 DWORD dwFlags,
                                                 DWORD renderopt,
                                                 ULONG cFormats,
                                                 DWORD* rgAdvf,
                                                 LPFORMATETC rgFormatEtc,
                                                 IAdviseSink* lpAdviseSink,
                                                 DWORD* rgdwConnection,
                                                 LPOLECLIENTSITE pClientSite,
                                                 LPSTORAGE pStg,
                                                 PVOID* ppv);
STDAPI Coign_OleCreateLinkFromData(LPDATAOBJECT pSrcDataObj,
                                                   REFIID riid,
                                                   DWORD renderopt,
                                                   LPFORMATETC pFormatEtc,
                                                   LPOLECLIENTSITE pClientSite,
                                                   LPSTORAGE pStg,
                                                   PVOID* ppv);
STDAPI Coign_OleCreateLinkFromDataEx(LPDATAOBJECT pSrcDataObj,
                                                       REFIID riid,
                                                       DWORD dwFlags,
                                                       DWORD renderopt,
                                                       ULONG cFormats,
                                                       DWORD* rgAdvf,
                                                       LPFORMATETC rgFormatEtc,
                                                       IAdviseSink*
lpAdviseSink,
                                                       DWORD* rgdwConnection,
                                                       LPOLECLIENTSITE
pClientSite,
                                                       LPSTORAGE pStg,
                                                       PVOID* ppv);
STDAPI Coign_OleCreateStaticFromData(LPDATAOBJECT pSrcDataObj,
                                                       REFIID riid,
                                                       DWORD renderopt,
                                                       LPFORMATETC pFormatEtc,
                                                       LPOLECLIENTSITE
pClientSite,
                                                       LPSTORAGE pStg,
                                                       PVOID* ppv);
STDAPI Coign_OleCreateLink(LPMONIKER pmkLinkSrc,
                                       REFIID riid,
                                       DWORD renderopt,
                                       LPFORMATETC lpFormatEtc,
                                       LPOLECLIENTSITE pClientSite,
                                       LPSTORAGE pStg,
                                       PVOID* ppv);
STDAPI Coign_OleCreateLinkEx(LPMONIKER pmkLinkSrc,
                                           REFIID riid,
                                           DWORD dwFlags,
                                           DWORD renderopt,
                                           ULONG cFormats,
                                           DWORD* rgAdvf,
                                           LPFORMATETC rgFormatEtc,
                                           IAdviseSink* lpAdviseSink,
                                           DWORD* rgdwConnection,
                                           LPOLECLIENTSITE pClientSite,
                                           LPSTORAGE pStg,
                                           PVOID* ppv);
STDAPI Coign_OleCreateLinkToFile(LPCOLESTR lpszFileName,
                                                 REFIID riid,
                                                 DWORD renderopt,
                                                 LPFORMATETC lpFormatEtc,
                                                 LPOLECLIENTSITE pClientSite,
                                                 LPSTORAGE pStg,
                                                 PVOID* ppv);
STDAPI Coign_OleCreateLinkToFileEx(LPCOLESTR lpszFileName,
                                                   REFIID riid,
                                                   DWORD dwFlags,
                                                   DWORD renderopt,
                                                   ULONG cFormats,
                                                   DWORD* rgAdvf,
                                                   LPFORMATETC rgFormatEtc,
                                                   IAdviseSink* lpAdviseSink,
                                                   DWORD* rgdwConnection,
                                                   LPOLECLIENTSITE pClientSite,
                                                   LPSTORAGE pStg,
                                                   PVOID* ppv);
STDAPI Coign_OleCreateFromFile(REFCLSID rclsid,
                                             LPCOLESTR lpszFileName,
                                             REFIID riid,
                                             DWORD renderopt,
                                             LPFORMATETC lpFormatEtc,
                                             LPOLECLIENTSITE pClientSite,
                                             LPSTORAGE pStg,
                                             PVOID* ppv);
STDAPI Coign_OleCreateFromFileEx(REFCLSID rclsid,
                                                 LPCOLESTR lpszFileName,
                                                 REFIID riid,
                                                 DWORD dwFlags,
                                                 DWORD renderopt,
                                                 ULONG cFormats,
                                                 DWORD* rgAdvf,
                                                 LPFORMATETC rgFormatEtc,
                                                 IAdviseSink* lpAdviseSink,
                                                 DWORD* rgdwConnection,
                                                 LPOLECLIENTSITE pClientSite,
                                                 LPSTORAGE pStg,
                                                 PVOID* ppv);
STDAPI Coign_OleLoadFromStream(LPSTREAM pStm,
                                             REFIID riid,
                                             PVOID* ppv);
STDAPI Coign_OleGetClipboard(LPDATAOBJECT* ppDataObj);
STDAPI Coign_CreateOleAdviseHolder(LPOLEADVISEHOLDER* ppOAHolder);
STDAPI Coign_OleCreateDefaultHandler(REFCLSID clsid,
                                                       LPUNKNOWN pUnkOuter,
                                                       REFIID riid,
                                                       PVOID* ppObj);
```

```
STDAPI Coign_OleCreateEmbeddingHelper(REFCLSID clsid,
                                                        LPUNKNOWN pUnkOuter,
                                                        DWORD flags,
                                                        LPCLASSFACTORY pCF,
                                                        REFIID riid,
                                                        PVOID* ppObj);
STDAPI Coign_OleRegEnumFormatEtc(REFCLSID clsid,
                                                 DWORD dwDirection,
                                                 LPENUMFORMATETC* ppenum);
STDAPI Coign_OleRegEnumVerbs(REFCLSID clsid,
                                           LPENUMOLEVERB* ppenum);
STDAPI Coign_OleCreateFontIndirect(LPFONTDESC lpFontDesc,
                                                   REFIID riid,
                                                   PVOID* ppv);
STDAPI Coign_OleCreatePictureIndirect(LPPICTDESC lpPictDesc,
                                                        REFIID riid,
                                                        BOOLEAN fOwn,
                                                        PVOID* ppv);
STDAPI Coign_OleLoadPicture(LPSTREAM lpstream,
                                          LONG lSize,
                                          BOOLEAN fRunmode,
                                          REFIID riid,
                                          PVOID* ppv);
STDAPI Coign_OleLoadPictureFile(VARIANT varFileName,
                                                LPDISPATCH* ppdispPicture);
STDAPI Coign_CreateStreamOnHGlobal(HGLOBAL hGlobal,
                                                   BOOLEAN fDeleteOnRelease,
                                                   LPSTREAM *ppStm);

///////////////////////////////////////////////////////////////// End of File.
#endif      // _COIGN_H_
```